# Automatic Test Generation for the Analysis of a Real-Time System: Case Study

Authors

Duncan Clarke

Insup Lee

Presents by
Mohammad Bawaneh

# Presentation Outline

- **Introduction.**
- **Objectives.**
- **Structure of Testing Methods.**
- **Approach.**
- **Case Study: The Philips Audio Control Protocol.**
  - **Formal Specification and Analysis.**
  - **Analysis through Testing.**
- **Conclusion.**

# Introduction

- **The Algebra of Communicating Shared Resources (ACSR) is a process algebraic formalism tailored to the domain of real-time systems.**

- **It relies on algebraic rewriting and state space exploration based verification techniques.**

- **Analysis real-time systems using ACSR can fall victim to state space explosion.**

# Presentation Outline

- Introduction.
- **Objectives.**
- Structure of Testing Methods.
- Approach.
- Case Study: The Philips Audio Control Protocol.
  - Formal Specification and Analysis.
  - Analysis through Testing.
- Conclusion.

# Objectives

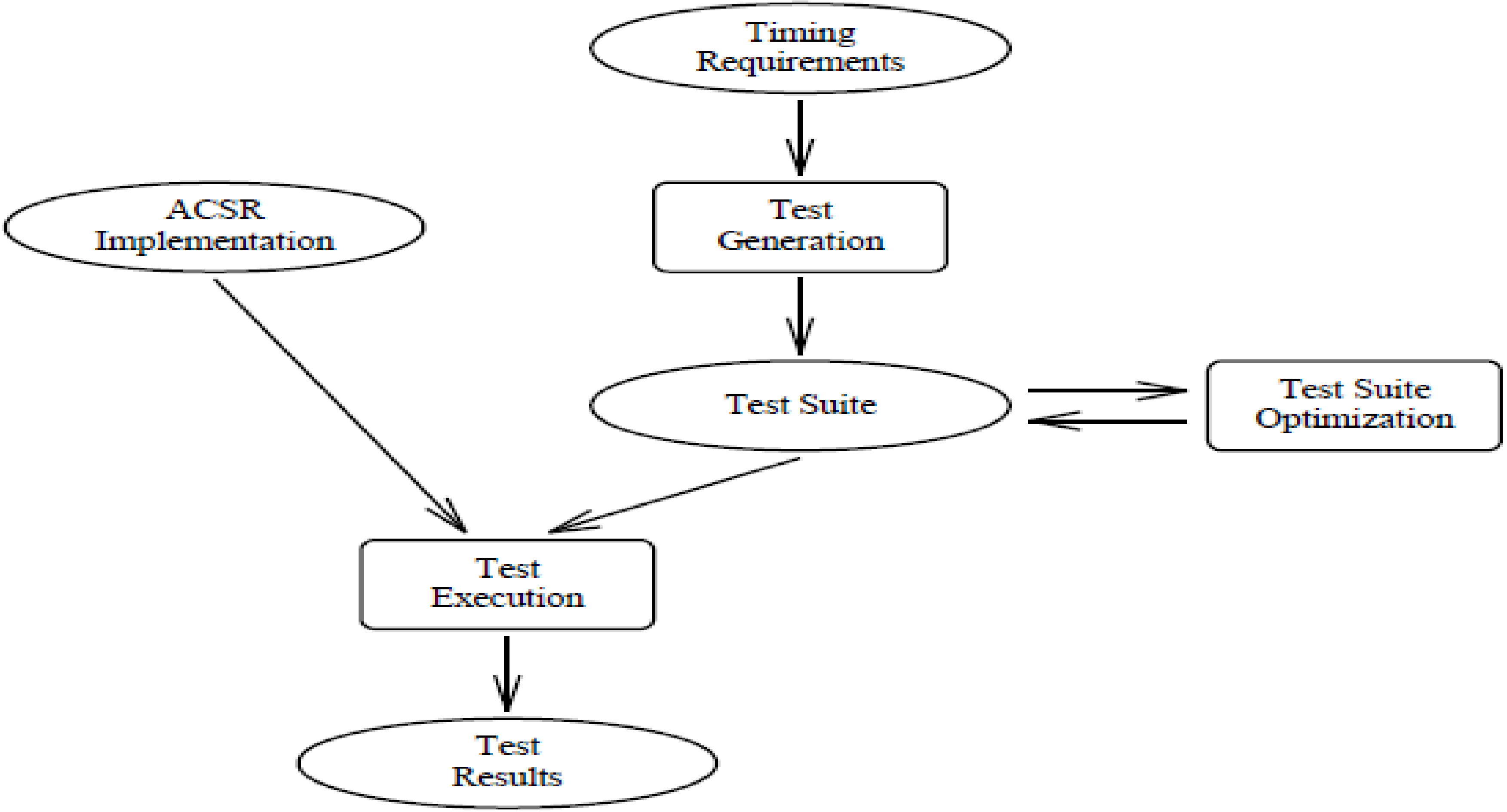
- **Present a testing approach which can be used to validate a design specification which has too many states.**

- **As an illustration of this benefit, authors describe the case study of using the automatic derivation of tests from timing specifications for the analysis of the Philips Audio Control Protocol.**

# Presentation Outline

- Introduction.
- Objectives.
- **Structure of Testing Methods.**
- Approach.
- Case Study: The Philips Audio Control Protocol.
  - Formal Specification and Analysis.
  - Analysis through Testing.
- Conclusion.

# Structure of Testing Methods



# Approach

- Introduction.
- Objectives.
- Structure of Testing Methods.
- **Approach.**
- Case Study: The Philips Audio Control Protocol.
  - Formal Specification and Analysis.
  - Analysis through Testing.
- Conclusion.

# Approach

- **Formal specifications needed to support the approach to testing formal models:**
  - **specifications for timing constraints to be tested.**
  - **specifications for the real-time system.**
  - **specifications for tests.**

# Timing Constraint Specifications

- **Timing constraints classified into:**
  - **Behavioral constraints.**

    limit the rate at which inputs are applied to a system.
  - **Performance constraints.**

    dictate the rate at which outputs are produced by a system.
- **In real-time systems such constraints are typically specified as a range of allowed delays between pairs of I/O events.**

# Timing Constraint Specifications

- In this testing framework, behavioral and performance constraints are described using the graphical constraint graph language.

- Constraint graphs reduce timing constraints to their most basic elements:
  - I/O events.
  - The delays that may elapse between them.
  - Timeout conditions.

# ACSR as a Test Language

- A test is an ACSR process that executes input events, output events, and time consuming actions.

- These events and actions represent the sequence of inputs and delays that make up the test, and the outputs that are expected in response.

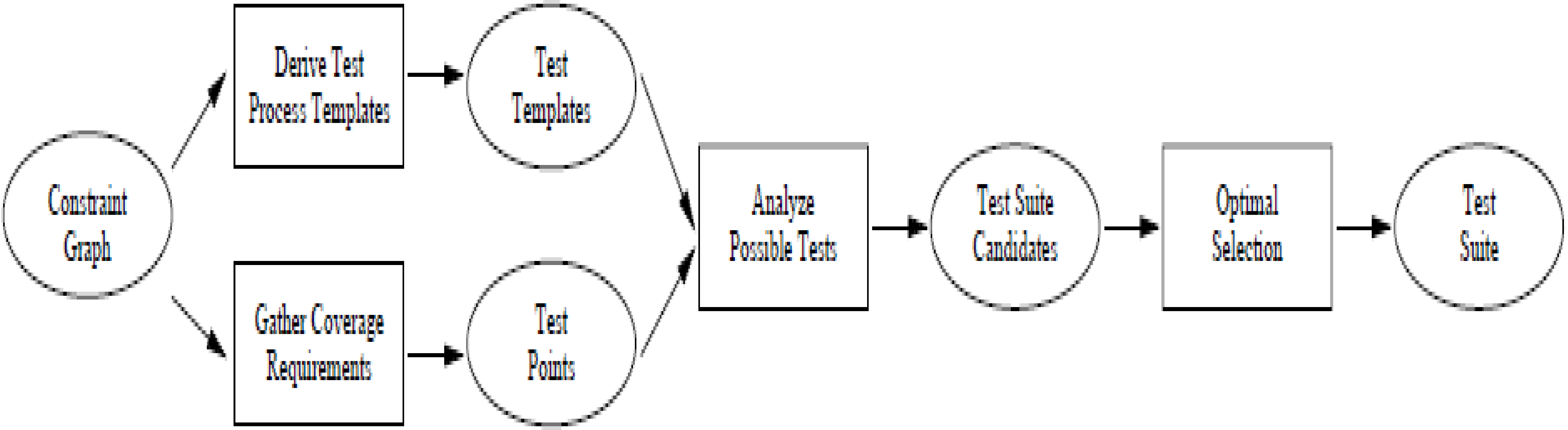
- Using ACSR's reactive semantics and scope operator to detect correct and incorrect sequences of outputs and delays generated by the test.

# Automatic Test Suite Derivation

- **The process by which constraint graph specifications of system constraints are translated to ACSR test suites.**

- **This test suits can be applied to a system to validate its conformance to the given timing specification.**

# Test Derivation Process



# Presentation Outline

- Introduction.
- Objectives.
- Structure of Testing Methods.
- Approach.
- **Case Study: The Philips Audio Control Protocol.**
  - **Formal Specification and Analysis.**
  - **Analysis through Testing.**
- Conclusion.

# Case Study: The Philips Audio Control Protocol

- It is the bus protocol that was used in Philips audio devices.

- The protocol is a data-link layer protocol for controlling communication between various components of an integrated stereo system (e.g., amplifier, tuner, CD player, etc.) in one physical unit.

# The Problem

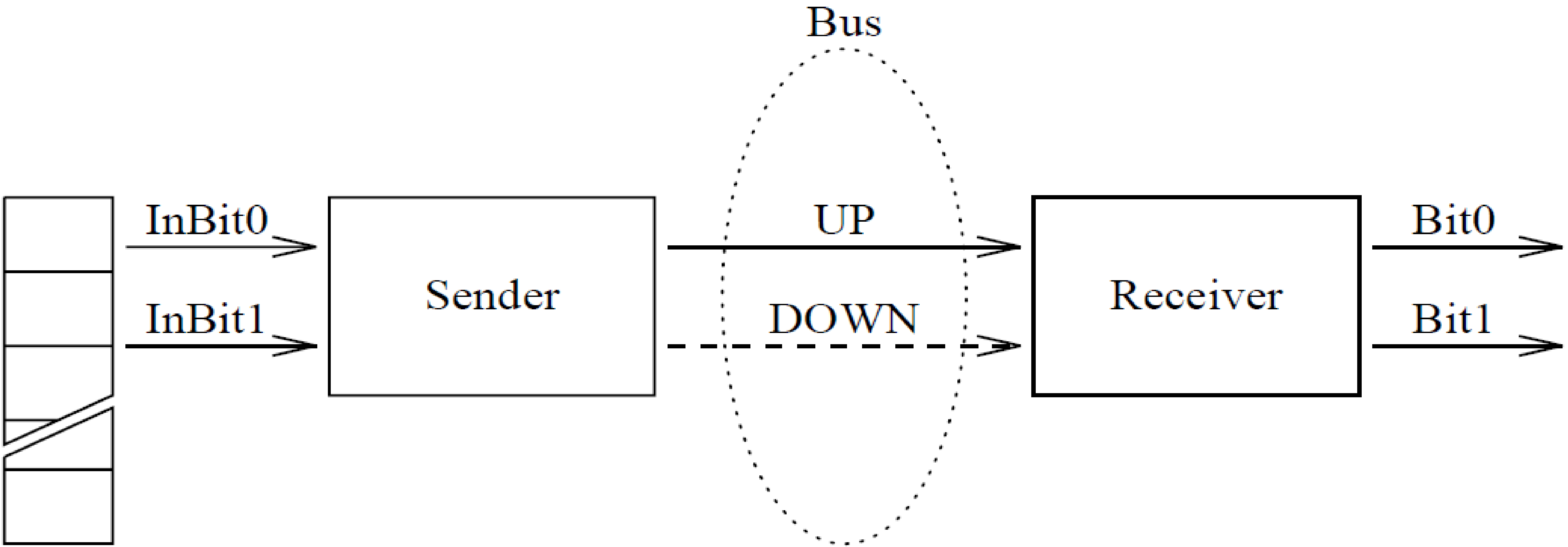
- **The novel aspect of the problem is that Philips has chosen an implementation that uses a few low-cost discrete components to realize the bus, and pre-existing microprocessor components to execute the LAN software.**

- **As a result there is the possibility of:**
  - **Drift between the clocks of the independent microprocessors.**
  - **Scheduling delays that arise from running on the non dedicated microprocessors.**

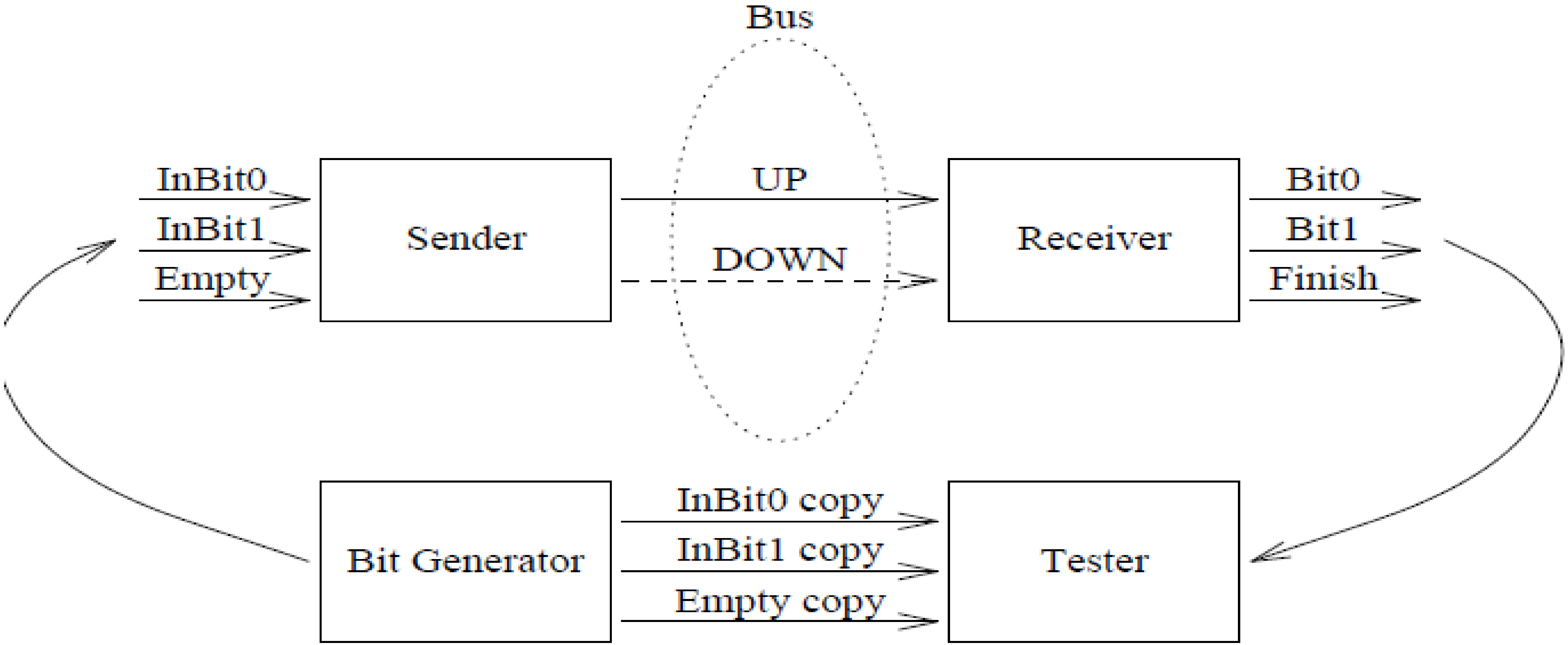
# The Problem

- The original specification of the Philips protocol allowed a tolerance of 5% on all timing.
- The traditional verification problem for this protocol has been to show that the specification is correct for this tolerance, and incorrect at some higher tolerance level.
- The protocol has a large number of states, which makes it challenging to carry out verification automatically.

# Components of Protocol System



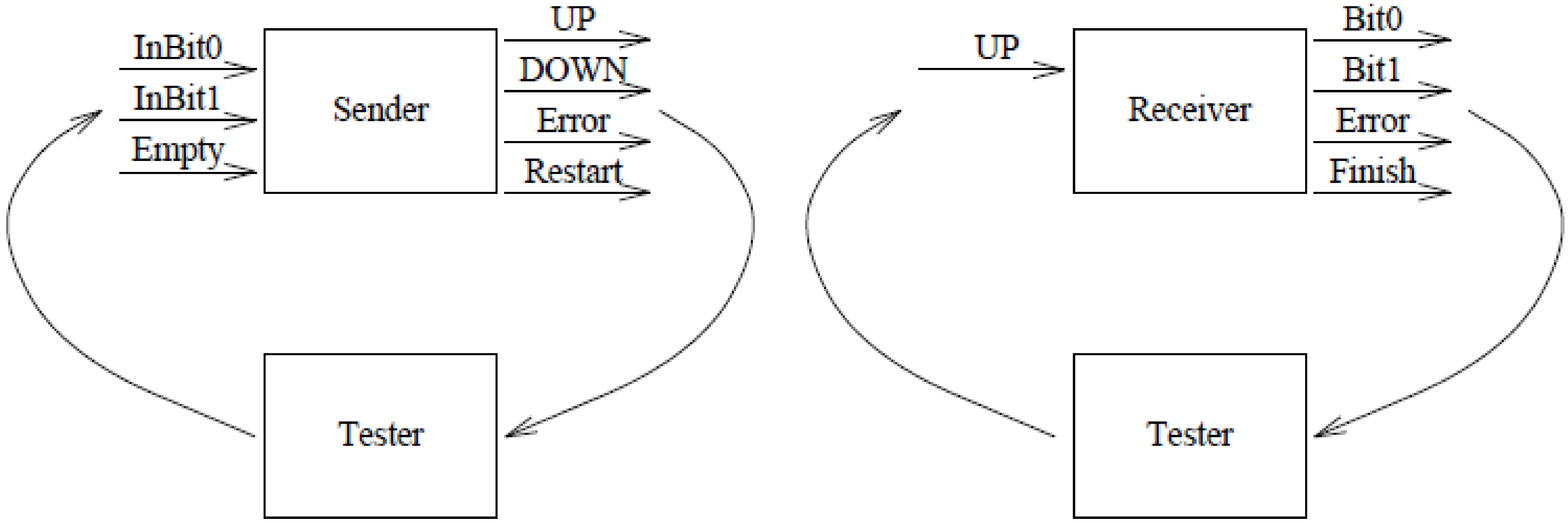
# Formal Specification and Analysis



# Empirical Results

- All verification at tolerance of 5% level was successful.
- All verification at tolerance of 6% level was failed.
- Varying time of bit slot were applied in this verification.
- However, experiment show that if the bit slot time was 888 microseconds, the memory resources required for verification at that precision exceed those available.
  - Memory required = 1000 MB.
  - Processor time for run (secs) = 10,000 secs.

# Analysis through Testing



# Empirical results at 5% level

- **All tests were capable of successful execution.**
- **A maximum of 225 MB of memory was required.**
- **A total of 2,365 CPU seconds elapsed ( for all tests).**

# Empirical results at 6% level

- **All tests were capable of successful execution and one test revealed as fault.**

- **A maximum of 230 MB of memory was required.**

- **A total of 3,144 CPU seconds elapsed (for all tests).**

# Presentation Outline

- Introduction.
- Objectives.
- Structure of Testing Methods.
- Approach.
- Case Study: The Philips Audio Control Protocol.
  - Formal Specification and Analysis.
  - Analysis through Testing.
- **Conclusion.**

# Conclusion

- A test derivation technique for verifying timing constraints on real time systems was presented.

- The testing method make it possible to analyze systems that are too large to verify using traditional state space exploration based approach.

# Thank You!